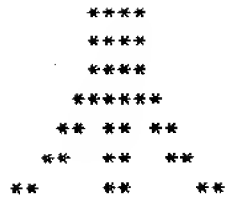
SCHEDULE B-1

Exhibit B



Atari Corporation

WORKING DRAFT

Atari TT and TT/X Product Specifications

The Design of the TT Computer Series

21 April 1989

Company Confidential
Trade Secrets Enclosed

Revisions

14 May 1988	Software contract compliance. Specification reflects final UXE design with a separate section listing engineering deviations in prototype printed circuit boards.
6 September 1988	Specification reflects the proposed TT architecture.
12 January 1989	Various small changes - ST-MFP moved before MFP-2 in memory map, floppy disk (3.2), and supplied current Ethernet strategy. Added cartridge port and sound to memory map.
23 January 1989	updates to DMA controller and music subsystem documentation further references added to bibliography
03 February 1989	correct DMA controller documentation
13 February 1989	prototype deltas; clarify low speed LAN; correct Microwire mask register address; add ikbd/MIDI ACIAs to I/O map; described optional workstation video card; corrected TT interrupt map IRQ5&6
03 March 1989	SCC clock specification; corrected ikbd and MIDI addresses; corrected DMA sound bit definitions; update cabinet notes
14 March 1989	clarify distinctions between TT and TT/X
29 March 1989	interrupt corrections software LAN selection configuration switch register location improve DMA sound documentation

References

The VMEbus is defined by:

(1) VMEbus International Trade Association (VITA), *VMEbus Specification Manual*, Revision C.1, October, 1985.

The Small Computer Systems Interface (SCSI) is defined by:

(1) Adaptive Data Systems, Inc., *SCSI Guidebook*, Issue Number 2, June, 1985.

Specific SCSI device implementation details for typical devices are available in:

(1) Adaptec Inc., *Description of SCSI Command Set for Communications Devices*, Revision 0.91, 1988.

(2) Archive Corporation, *VIPER Product Manual; SCSI Models 2060S and 2150S*, Part No. 21391-001, June, 1988.

(3) Maxtor Corporation, *XT-4000S OEM Manual & Product Specification*, 1014995, 1987.

(4) Quantum Corporation, *Q200 Series Programmer's Manual*, 81-45416, Rev. B, 1987.

Details of the major commercially available chips used in the TT/X architecture are contained in:

(1) General Instrument Corp., *AY-3-8910/8912 Programmable Sound Generator Data Manual*, February, 1979.

(2) Logic Devices Inc., *L5380/L53C80 CMOS SCSI Bus Controllers*, September 1988

(3) Motorola, Inc., *MC68030 Enhanced 32-Bit Microprocessor User's Manual*, 2nd. Edition, 1989.

(4) Motorola, Inc., *MC68881/MC68882 Floating-Point Coprocessor User's Manual*, First Edition, 1987.

(5) Motorola, Inc., *MC68901 Multi-Function Peripheral*, January, 1984.

(6) Motorola, Inc., *MC146818A Real-Time Clock Plus RAM (RTC)*, 1984.

(7) Motorola, Inc., *MC6850 Asynchronous Communications Interface Adapter*, 19??.

(8) Western Digital Corp., *WD1772-02 Floppy Disk Formatter/Controller*

(9) Zilog, Inc., *Z80C30 CMOS Z-BUS SCC / Z85C30 CMOS SCC Serial Communications Controller - Preliminary Product Specification*, October, 1987.

(10) Zilog, Inc., *Z80C30 CMOS Z-BUS SCC / Z85C30 CMOS SCC Serial Communications Controller - Technical Manual*, September, 1986.

The ST compatible hardware interfaces are also described in:

(1) Atari Corporation, *Engineering Hardware Specification of the Atari ST Computer System*, January 7, 1986.

(2) Atari Corporation, *ST DMA Sound Technical Reference*, July 18, 1988.

(3) Atari Corporation, *Intelligent Keyboard (ikbd) Protocol*, February 26, 1985.

(4) Atari Corporation, *[ST] DMA Controller*, September 26, 1984.

(5) Atari Corporation, *Atari ACSI/DMA Integration Guide*, January 23, 1989.

The TT boot procedure and disk partitioning are described in the following documents:

(1) Atari Corporation, *Atari TT Boot Specification*, Alan Char, October 27, 1988.

(2) Atari Corporation, *Atari ST Disk Partition Map*, Memorandum by Minna Lai, December 8, 1988.

(3) IBM, *Disk Operating System, Version 3.30 Technical Reference*, 1st Edition, April 1987.

The Binary Compatibility Standard is defined in:

(1) Motorola Inc./UniSoft Corp.,*M68000 Binary Compatibility Standard*, Draft 11, November 10, 1988.

Block Diagram

Block Diagram Goes Here!

TABLE OF CONTENTS

1. INTRODUCTION

The TT (Thirty-two/Thirty-two bit) and TT/X (Unix[1] Engine) are a new series of Atari computers designed as enhanced versions of the existing ST and MEGA family. The TT series maintains compatibility with the ST/MEGA architecture, but uses the Motorola 68030 microprocessor and provides enhanced graphics and sound. The TT is also designed to allow it to run UNIX, without any speed penalty caused by ST compatibility constraints.

The TT/X is an up-market derivative and superset of the TT, intended primarily for the UNIX market. Whereas the TT is a desktop product, the TT/X is packaged for desk side use. In this document, and from an architectural standpoint, the TT and TT/X are identical. Only packaging differentiates the two products.

The TT series are based around the high performance 32-bit Motorola MC68030 processor running at a 16 MHz clock frequency. The 68030 includes on-chip data and instruction caches which can be filled from some regions of memory in bursts of double word fetches.

The architecture also includes the industry standard VMEbus to facilitate expansion. The system supports the latest revision (C.1) of the VMEbus specification. The TT can accommodate one single-Eurocard VME board, whereas TT/X is designed for double-Eurocards.

In the TT/X, special attention has been paid to provide the hardware support necessary to permit multiprocessing. These features could be used in the future to provide a still higher performance system, either by adding additional similar coprocessors or adding a very high speed replacement processor that could still take advantage of the existing I/O subsystem.

The TT series is expected to function in an environment with other TTs and even machines from different manufacturers. To facilitate connectivity, each system has an on-board port that for a moderate speed LAN. If the LAN is not being used, the port can be programmed to be a standard RS232C port. Through an optional VMEbus-based Ethernet controller, the TT/X will also have the capability of connecting to heterogeneous Ethernet networks. Additionally, each TT has three standard RS-232C serial ports for connection to modems, display terminals, or digitizing tablets.

The TT is intended for use with either TOS or the UNIX operating system. The TT/X's principle operating system is UNIX. The initial product offering is based around UniSoft's UniPlus+ V Release 3 version 1 which is fully compatible with AT&T System V Interface Definition (SVID), the Portable Operating System Interface Specification (POSIX), and the X/Open Portability Guide. The X Window System, a network transparent window system originally developed at MIT, will also be available in the initial release. A windowing user interface running on top of X will be provided.

---

[1] Unix is a trademark of AT&T.

The hardware specifications of the TT series of computers is as follows:

- Motorola MC68030 at 16MHz
- Motorola MC68881/68882 Floating Point Coprocessor (optional/socketed)
- RAM: 2 Mbyte of dual-purpose (video/system) RAM, expandable by an add-on daughterboard containing a further 2 or 8 Mbyte of dual-purpose memory. This memory appears 64-bits wide to the video logic and 32-bits wide to the rest of the system. TT video logic must have access to this memory on a time critical basis. The remaining system logic, including the processor, can access this memory in the alternate 250 nS time slices.
- RAM: 4 Mb nibble-mode memory daughter-board(s) allowing another 4Mb (TT) or 16Mb (TT/X) expansion.
- ROM: 4 socketed 1 Mbit ROMs, providing 512Kb of ROM space. All four ROMs must be present, because of the 32-bit wide system bus access.
- internal video modes that are a superset of those in the Atari ST series-- Color: 320x200, 320x480, 640x200, 640x480. DuoChrome: 640x400. Monochrome: 1280x960.
- an industry standard analog RGB color monitor interface (for color and DuoChrome modes)
- a high performance ECL monitor interface (for the high resolution monochrome mode)
- parallel I/O port, implemented using the one of the parallel ports on the General Instruments AY-3-8910 / Yamaha YM-2149 sound chip
- internal speaker, that can be disabled under software control
- 2 low-speed async serial I/O ports (one from each of two 68901 MFPs)
- 2 high-speed SDLC serial I/O ports (from a Zilog 8530 SCC), one port of which can be strapped to be a LAN interface with a proprietary single channel DMA controller
- real time clock (RTC) with 50 bytes of non-volatile RAM
- ST/MEGA compatible intelligent keyboard, with mouse and joystick ports
- Atari ACSI DMA channel (for Atari Hard Disk, Laser Printer, CD-ROM, etc)
- floppy disk controller and interface sharing the ACSI DMA channel
- Musical Instrument Digital Interface (MIDI)
- Atari ST compatible cartridge port (128 Kbyte storage)
- SCSI interface with a 25-pin connector implemented with the NCR 5380 SCSI controller chip and a proprietary DMA controller

- VMEbus for expansion: TT contains 1 single Eurocard A24/D16 slave-only interface, the TT/X allows 5 slots for full multi-master VME with the address space divided into A32/D32, A24/D16, A16/D16

2. *MAIN SYSTEM*

The TT series architecture is designed to be a high performance computing platform. By including the VMEbus and facilities for multi-processing the system can be expanded for future needs.

2.1. *Processor and MMU*

The TT uses the Motorola MC68030[2] 32-bit microprocessor. This single chip contains a 68020 superset processor, a paged memory management unit, and independent instruction and data caches. The 68030 is a complex instruction set computer (CISC) that extends the 68000 instruction set and enhances the addressing modes. The processor will be clocked at 16.1 MHz.

The MMU in the 68030 is a subset of that provided by the Motorola MC68851. In particular, the translation look-aside buffer (TLB) has been reduced to 22 entries, requiring particular care in memory assignment to avoid unnecessary descriptor thrashing.

The on-chip instruction and data caches maximize processor throughput while reducing the bus bandwidth necessary to fuel the processor.

2.2. *Floating Point Coprocessor*

The TT design has a socket for an optional the Motorola MC68881 or the newer, higher-performance, MC68882. The MC68881 will be a standard feature on the TT/X. These two parts are hardware compatible. There is a slight software difference in the size of the exception stack frames, but it is possible to write software that will run transparently with either part.

The floating point operations are performed in accordance with IEEE Standard 754, with both 32-bit (single) and 64-bit (double) precision external access.

The floating point coprocessor is run at the same clock speed as the main processor. It appears as the "standard" floating point coprocessor ID of 1 in the 68030 CPU address space.

2.3. *ROM*

The system includes on-board sockets for a set of four 1Mbit ROMs, providing a total of 512Kb ROM. Since system bus access is 32-bits wide, all four ROMs must be present. Jumpers are provided to allow the use of 27256, 27512, 27010/27C1001, and 57101/27C1000 EPROMs, in addition to 53100 ROMs. The default jumper position allows the use of 27512 EPROMs (for a total of 256 Kb of ROM) as well as 571001/27C1000 EPROMs or 531000 ROMs (for a total of 512 Kb of ROM). 32 pin sockets are

---

[2] MC68030, MC68020, MC68851, MC68881, and MC68882 are trademarks of Motorola, Inc.

provided, although 27256, 27512, and 531000 only use the bottom 28 pins.

An image of the first 8 bytes of ROM resides in the first 8 bytes of the ST compatible image. These first 8 bytes (0x00000000-0x000007, or 0xFF000000-0xFF000007 in the image) are accessible *only* in supervisor mode. Attempts to read from this area in user mode or any write results in a bus error. A VMEbus master would have to do a privileged accesses to read the ROM at these locations. The full ROM resides at the memory location 0x00E00000 - 0x00EFFFFF (with an image at 0xFFE00000 - 0xFFEFFFFF).

Among the tasks this ROM perform are system initialization, power-on diagnostics, and boot code that can boot from a floppy, ACSI device, SCSI device, or network. The ROM is expected to contain a multi-lingual implementation of TOS. Moreover, if sufficient space is available, ROM-based service diagnostics will be provided.

2.4. RAM

The basic system includes 2 Mbytes of dual-purpose RAM which is used for both video and system memory. This is implemented by using 16 256Kbitx4 100 nS DRAMs, yielding a 64-bit wide internal bus for high performance video access.

The bus architecture is similar to the ST in that memory access cycles are interleaved between the MPU and the video controller in 250 nS RAM time slices, thus allowing video display memory to reside efficiently as part of main memory. During active display cycles the processor is prevented from accessing the memory but is allocated the next 250 nS time slice. The processor interfaces to this RAM through a 32-bit bus, but the video subsystem itself accesses memory on a 64-bit wide bus. The video chip (TT shifter) has on-chip buffering to provide very high bandwidths for data.

A pin on the memory controller can be strapped to force the memory controller to ignore the configuration register and automatically use 256K part mode. This is used for the (optional) second of two memory controllers. The configuration register still applies to the primary memory controller. This allows 2 Mbyte or 8 Mbyte connected to the primary memory controller, and 2 Mbyte to the secondary controller for possible dual-purpose RAM configurations of 2, 4, 8, or 10 Mbyte.

Single-purpose RAM daughter-boards are possible as an option. By eliminating the video timing constraints on this RAM, this memory can be made to appear faster to the processor. The daughter boards are currently implemented by using 32 1 Mbit 100 nS DRAMs. When 4 Mbit DRAMs become available, it will be possible to provide 16 Mbyte of single-purpose RAM on a single daughter card. The single-purpose memory system uses nibble mode RAMs to facilitate burst mode filling of the 68030 caches.

Additional memory can be installed in the system by plugging in VME memory cards. If A32/D32 cards are used, the VME RAM will be contiguous with the daughter-board single-purpose RAM. The VME RAM cards will run slightly slower than the system RAM as all VME accesses incur an extra

wait state per bus cycle.

The MC68030 accesses to on-board RAM typically require 4 clock cycles.

There is no provision for parity or ECC protection on the system RAM. The reliability of current DRAM technology makes this unnecessary. However, such features could be included in add-on VME cards.

The local RAM on the system board is accessible from the VMEbus as bytes, words, or double words.

The first 0x800 bytes (2K) of RAM (0x00000008-0x000007FF, or 0xFF000008-0xFF0007FF in the image) are accessible *only* in supervisor mode. Attempts to read or write to this area in user mode results in a bus error. VMEbus masters must do privileged accesses to use this RAM.

The memory refresh period is programmable by writing don't care data to an address in the range 0xFFD00000 - 0xFFD000FF. The least significant byte of the address sets the number of system clock cycles between each refresh request (writing to 0xFFD00000 stops memory refresh). Writing to 0xFFD00001 sets the fastest refresh rate, while writing to 0xFFD000FF sets the slowest refresh rate. If the refresh rate is set too fast, the processor will never be granted access to RAM since refresh has priority. This implies that the refresh control count should never be set less than 0x08 (by writing to 0xFFD00008). This value will typically be set to a value that is a function of the system clock frequency during the system initialization performed by the boot ROM, and then left alone. The actual process of doing a write to this region will simultaneously cause a BUS ERROR, which should be ignored. (Note that if a poorly behaved program writes to physical locations in the 0xFFD00000 - 0xFFD000FF range, the bus error handler should be prepared to reset the refresh rate to a reasonable value.)

2.5. *System Control Unit*

The System Control Unit (SCU) provides an additional level of interrupt control for the system. It also contains registers that allow the software generation of interrupts. All of the SCU registers are reset at power-on and by the reset pushbutton.

2.5.1. *Interrupt Mask and Current Status*

The SCU contains two mask registers that permit independent control over which interrupt levels will be seen by the processor. One register masks interrupts generated on the system board and the other masks VMEbus sources. These registers are cleared at power-up or reset, disabling all interrupts.

There are also interrupt request registers that show the current state of the seven interrupt request levels from each of the sources. This register shows the physical status of the interrupt lines before they are ANDed with the SCU's mask register.

The motherboard sources for IRQ5 and IRQ6 can be serviced by either the 68030 or the VMEbus master. The implementation used means that IRQ5

and IRQ6 look to the 68030 like VME interrupts, and can not be masked independently with the SCU motherboard interrupt mask register.

2.5.2. System Control Registers

The SCU also contains two read/write registers that can be used for system configuration information.

2.5.3. Interrupt Generator

The system can write to an I/O address to generate a low priority (level 1) interrupt to the 68030. This I/O address contains a read/write status/control port, only the least significant bit of the least significant byte is defined. When set to 1, it generates an auto-vectored level 1 interrupt. When cleared, the interrupt request is taken away.

The SCU is hardwired so that:

- only interrupts 5 and 6 have external IACK pins and are capable of generating vectored interrupts on the motherboard (and also cause VME IRQ5 and IRQ6 respectively)
- SCU generated IRQ1 and IRQ3 are hardwired to the corresponding priorities and are always autovectored
- SCU generated IRQ1 is detected only by the MPU not the VMEbus
- VMEbus ACFAIL generates a system (motherboard) IRQ7 to the MPU, but does not generate an IRQ7 to the VMEbus. The only other source of an IRQ7 is a VMEbus card.

2.5.4. Bus Timer

The SCU also implements a system bus timer. If nothing concludes a bus cycle within 16 microseconds, the SCU will signal a bus error.

2.6. DMA Controllers

The TT series includes three independent DMA channels: 1) the low speed network port implemented on SCC serial port A, 2) the SCSI port and 3) the ST "ACSI"/Floppy DMA. Additionally, the VMEbus interface permits a VMEbus master to perform DMA into system memory. The following is the DMA bus mastership priorities:

priority	function
highest	ACSI/Floppy Controller
	SCC DMA Channel
	SCSI DMA Channel
	VMEbus Masters
lowest	68030

2.6.1. SCC and SCSI DMA Channels

The SCC and SCSI DMA controllers assemble the bytes from the peripheral into double words for writing to the system bus. This feature is actually implemented with two independent "assembly" double words so that when one has been filled and is waiting for access to the processor

bus, the second can be filling. If the second assembly word fills before the bus is released by the DMA chip, it will be written in the same bus transaction.

DMA can be done to any byte boundary of any double word wide memory space, either on the main system board or on the VMEbus. DMA is done in the *physical* address space.

The programmer's model of each of these DMA channel consists of:

- a word wide read/write status/control register that contains direction, enable and bus error bits
- four bytes forming a 32-bit DMA pointer,
- partial input register that must be read and merged with RAM contents under CPU control if the DMA input is done to a point in RAM that is not on a double word boundary or if DMA is not done in multiples of four bytes,
- a 32-bit wide DMA byte count (implemented in four separate bytes).

A DMAC controller exists for each channel: SCC and SCSI. Each DMA controller is physically implemented in two chips: one for the system bus interface, one for peripheral interface and FIFO. The bus interface controller is strapped externally for either SCSI or SCC.

The software that sets up the DMAC for DMA transfers must account for the DMAC being a byte-wide peripheral appearing on the odd bytes of the address bus. This requires the 68030 either to use the MOVEP instruction or to do rotates and four separate byte output operations to put out a 32-bit address or byte count.

DMA Controller Registers

offset	width	function
0x00	OB	DMA Pointer Upper
0x02	OB	DMA Pointer Upper-Middle
0x04	OB	DMA Pointer Lower-Middle
0x06	OB	DMA Pointer Lower
0x08	OB	Byte Count Upper
0x0A	OB	Byte Count Upper-Middle
0x0C	OB	Byte Count Lower-Middle
0x0E	OB	Byte Count Lower
0x10	W	Data Residue Register High
0x12	W	Data Residue Register Low
0x14	OB	Control Register

The control word is a bit-mapped register:

bit	function
0	DMA Direction Out (1 = out to port)
1	Enable (0 = off, 1 = on)
2-5	<reserved>
6	Byte Count Zero (1 = terminal count)
7	Bus Error (1 = Bus Error occurred during DMA by this channel)

To perform DMA:

1) set the DMA controller direction
2) set the base address
3) set up the peripheral for DMA
4) then set the enable bit

The direction and enable bits should not be set in the same operation.

If DMA input is done to anything but a double word aligned destination, or if the length is not a multiple of 4, the final byte(s) of the transfer will not be written to the system RAM. It is then the programmer's responsibility to read the Data Residue Register and merge the input with the contents of the appropriate double word in RAM. (The least significant two bits of the DMA pointer are correctly incremented, which can be used to determine how much of the Residue Register is valid.)

DMA can only be done to double word width ports, like RAM and D32 VME cards.

If an attempted DMA operation generates a bus error, the DMA operation is immediately disabled and the bus error bit set in the Control/Status register. The bus error status bits of each of the DMA controllers routed to individual MFP-2 input bits where they can be read or optionally used to generate an interrupt. The bus error status for a channel is automatically cleared by reading the channel's control register.

The DMA byte count register generates an interrupt when the byte count reaches 0. The DMA is automatically disabled by reaching the terminal count.

The NCR 5380 SCSI Interface Chip must not be used in BLOCK MODE DMA for use with the TT DMA controllers. The SCC should be in programmed to use the WAIT/*REQ pin in *REQ mode when doing DMA.

2.6.2. Floppy/ACSI Interface

The ST compatible Floppy/ACSI subsystem interfaces between dual-purpose RAM and ACSI compatible peripherals, such as the SLM804 laser printer, SHxxx/Megafile hard disk drives, and Atari CD-ROM. This DMA channel is shared with the internal floppy disk controller.

DMA between RAM and ACSI peripherals, and between RAM and floppy, can only be performed using the dual-purpose RAM. If a transfer is required from such a device into standard ("single-purpose") system RAM, a two stage transfer is required, using the dual-purpose RAM as an intermediate buffer.

2.7. Real Time Clock

The TT system includes a Motorola MC146818A Real Time Clock chip. This provides time of day (down to one second resolution), date, and a programmable periodic interrupt. The RTC is provided with a 32.768 kHz oscillator that is independent of all other system clocks.

The interrupt output of the real time clock chip connects to one of the MFP parallel inputs.

The chip also includes 50 bytes of battery backed up (non-volatile) RAM that is used for storing diagnostic and configuration data.

The chip is accessed through two consecutive word ports. The first word is a write-only port that is used to set the real time clock chip address that is desired. The second word is the read/write data port. When doing a write to a clock chip register, it is possible to do a double word write; the first word would set the address, and the second word the data.

3. *Device Subsystems*

The TT architecture supports the following device subsystems:

- SCSI (as defined by the ANSI X3T9.2 committee)
- ST compatible ACSI
- floppy disk interface sharing the ST "ACSI" DMA channel
- high-speed serial ports and a low speed network port through the SCC chip
- two additional serial ports and an external interrupt port connected to MFP controllers
- a Centronics parallel printer port driven by the Yamaha YM-2149 sound chip
- a ST/MEGA compatible intelligent keyboard, mouse, and joystick interface
- a port supporting application and diagnostic cartridges

3.1. SCSI

The TT implements the complete single-ended (non-differential) SCSI bus by using the NCR5380 SCSI Controller. The NCR5380 is used in 8-bit asynchronous data transfer up to 4.0 Mbytes/second, adequate for current disk drives.

The SCSI connector provides for connection of SCSI compatible devices through a 25-pin D connector. Internally, the full 50-pin cabling is used.

In a typical configuration, the SCSI bus will be used to provide the main mass storage elements of the system. For the TT/X a SCSI hard disk of at least 40 Mb (formatted) will be used, with typical software development stations using considerably larger disks. The SCSI bus can also be used for removable media devices such as the Syquest cartridge drives and magnetic tape controllers. The default system hard disk will be SCSI unit 0, device 0.

The SCSI bus can support up to 7 major devices (in addition to the TT itself).

3.2. ACSI

3.3. Floppy Disk

The TT series floppy disk subsystem is designed around the WD1772 Floppy Disk Controller supporting up to two daisy-chained floppy disk drives (drive 0 or 1). A higher speed version of the 1772 is planned to allow 1.44Mb (formatted) capacity drives. The TT is designed for one internal floppy disk drive and one external drive (such as the SF314). The TT/X can have two internal floppy disk drives.

The subsystem interfaces to the dual-purpose RAM through the ACSI DMA controller. Command and arguments are sent to the FDC by first writing to the DMA Mode Control Register to select the desired FDC register and then writing the data bytes.

The standard floppy for the TT series is the 3.5 inch floppy disk with the capacity of 720 Kbyte (formatted). The 1.44Mb drives will be available as an option.

The internal drive cabling supports the DiskChangeLine signal from the floppy drive(s) to a bit on MFP-2. DiskChangeLine can be read when the drive is selected, and is asserted when (1) power is applied or (2) a diskette is removed from the drive. The signal is cleared by issuing a step command to the drive.

3.4. High Speed Serial Ports

The Zilog 85C30 SCC, a dual channel, multi-protocol data communications peripheral, is included in the TT design to provide two serial ports (ports A and B).

Port A can be used as either a network port or a standard low speed RS232C port. When bit 7 of the GI Sound Chip port A is a 0, LAN mode is selected. The input/output of Port A is routed to the appropriate connector: (1) if RS232C mode is selected, the port is connected to a DB-9P or (2) if the network port is selected, it is connected to an 8-pin mini-DIN connector. The output pins on the unselected port remain inactive.

The SCC handles both asynchronous formats and synchronous byte-oriented protocols such as HDLC and IBM's SDLC.

Port B is configured to be a low speed RS232C serial port that can be used for connecting to a modem or a local mainframe. It is pinned out on a DB-9P connector in a way that is compatible with the Atari PC4. Modem control signals are derived directly from the 85C30 port B control lines. This port can operate with split transmit and receive baud rates.

The PCLK input to the SCC is 8 MHz. The RTxCA input is provided with a 3.672 MHz clock. The input to TRxCA comes from the low speed LAN connector. RTxCB is run at 2.4576 MHz. TRxCB is generated by the Timer C output of the second (TT) MFP.

3.4.1. SCC RS232 Port Pinout

The SCC RS232 serial ports are pinned out in DB-9P connectors in a way that is compatible with the Atari PC4. On the TT, the SCC port A RS232 connections are routed to a header on the motherboard. That header can be connected with a ribbon cable to a nine pin D connector located on the VME slot cover.

SCC RS232 Pinouts

pin	Port A (RS232 Mode)	Port B
1	Carrier Detect (I)	Carrier Detect (I)
2	Receive Data (I)	Receive Data (I)
3	Transmit Data (O)	Transmit Data (O)
4	Data Terminal Ready (O)	Data Terminal Ready (O)
5	Ground	Ground
6	Data Set Ready (I)	Data Set Ready (I)

7	Request to Send (O)	Request to Send (O)
8	Clear to Send (I)	Clear to Send (I)
9	--	Ring Indicator (I)

Note: The SCC Port B Ring Indicator (RI) signal is connected to bit 6 of the MFP-2 General Purpose I/O Port (GPIP).

3.4.2. LAN Connector Pinout

The moderate speed LAN connector is an 8 pin female mini-DIN.

SCC LAN Pinout (Port A)

pin	function
1	Output Handshake (DTR, RS423)
2	Input Handshake/External Clock
3	Transmit Data -
4	Ground
5	Receive Data -
6	Transmit Data +
7	<reserved>
8	Receive Data +

3.5. MFP

Two 68901 Multi-Function Peripheral (MFP) controllers are used to provide system timers, low speed RS232C serial ports, and an interrupt controller. One MFP, designated MFP-ST, is used in a way that is compatible with the ST. It provides both a serial port and interrupt control. A second MFP provides another low speed serial port and more I/O and interrupt pins.

The baud rate clock for the MFPs serial transmitter and receiver is derived from the timer D output of each MFP. Given the MFPs' 2.4576 MHz clock, baud rates up to 19.2 Kbaud can be supported on these serial ports.

3.5.1. MFP Serial Port Pinouts

Both MFP serial ports are pinned out in DB-9P connectors in a way that is compatible with the Atari PC4. On the TT, the MFP-2 serial port is routed to a header on the motherboard. That header can be connected with a ribbon cable to a nine pin D connector located on the VME slot cover.

One of the MFP serial ports has a complete complement of modem control lines compatible with the ST, but pinned out in a 9 pin D connector. The other MFP serial port provides only a "three-wire" interface.

MFP Serial Port Pinouts

pin	MFP-ST	MFP-2
1	Carrier Detect (I)	--
2	Receive Data (I)	Receive Data (I)
3	Transmit Data (O)	Transmit Data (O)
4	Data Terminal Ready (O)	--
5	Ground	Ground

6	--	--
7	Request to Send (O)	--
8	Clear to Send (I)	--
9	Ring Indicator (I)	--

Note: The Ring Indicator (RI) signal is connected to bit 6 of the MFP-ST General Purpose I/O Port (GPIP).

3.5.2. Uncommitted I/O Pins

The least significant two bits of MFP-2's General Purpose I/O Port are not currently used and are routed to a dual row of stakes for convenience. These are simple unbuffered TTL level signals that can be used for either input or output.

3.6. Parallel Printer Port

The TT architecture includes a bi-directional 8-bit parallel printer port that implements a subset of the Centronics standard. This interface is through the General Instruments AY-3-8910 / Yamaha YM-2149 Programmable Sound Generator (PSG) chip. It is pinned out in a DB255 in a way that is a subset of the Atari PC4. The Centronics STROBE signal is generated from a PSG bit. The Centronics BUSY signal from the printer connects to one of the parallel input lines of the MFP to permit interrupt driven printing. Eight bits of read/write data are handled through I/O port B on the PSG at a typical data transfer rate exceeding 4000 bytes/second.

3.7. Keyboard Interface

The TT keyboard interface is completely compatible with the ST/MEGA computers. The keyboard is equipped with a combination mouse/joystick port and a joystick only port. The keyboard transmits encoded make/break key scan codes (with two key rollover), mouse/trackball data, joystick data, and time-of-day. The keyboard receives commands and sends data via bidirectional communication implemented with a MC6850 Asynchronous Communications Interface Adapter (ACIA). The data transfer rate is 7812.5 bits/second. All keyboard functions, such as key scanning, mouse tracking, command parsing, etc. are performed by a HD6301V1 8-bit microcomputer unit. (See the Atari, *Intelligent Keyboard (ikbd) Protocol*, February 26, 1985.)

3.7.1. Mouse and Joystick Interface

The Atari two-button mouse is a mechanical, opto-mechanical, or optical mouse with the following minimal performance characteristics: a resolution of 100 counts/inch, a maximum velocity of 10 inches/second, and maximum pulse phase error of 50%. The joystick is a four direction switch-type joystick with one fire button.

3.8. ROM Cartridge

The TT's cartridge port is fully compatible with ST cartridges. The cartridge is physically connected through a 40 pin card edge connector ROM cartridge slot. Cartridge ROMs are mapped to a 128K memory region starting at 0x00FA0000, extending to 0x00FBFFFF (with an image at

0xFFFA0000 to 0xFFFBFFFF).

4. *Video Subsystem*

The TT video subsystem is designed to extend the existing ST modes. Additional modes are available on the TT that allow more colors and larger screen sizes. This subsystem one of the basic components required to support the industry standard X Windows windowing system allowing the TT to exist as a fully-compatible X Windows workstation.

4.1. *Video Configuration*

The various modes available on the TT are:

ST mode

mode bits	resolution	planes	palette (CLUT entries)	colors DACs
00	320x200	4	16	512/3-bits
01	640x200	2	4	512/3-bits
10	640x400	1	-	Monochrome

TT mode

mode bits	resolution	planes	palette (CLUT entries)	colors DACs
000	320x200	4	16	4096/4-bits
001	640x200	2	4	4096/4-bits
010	640x400	1	2	4096/4-bits (Duochrome)
100	640x480	4	16	4096/4-bits
110	1280x960	1	-	Monochrome
111	320x480	8	256	4096/4-bits

As the table indicates, the modes are set through either the respective (ST or TT) Shift Mode Register. In the ST mode, 16 word-wide registers comprise the ST Color Palette (also known as the Color LookUp Table - CLUT). Contained in each entry are nine-bits of color: 3-bits each for red, green, and blue. Therefore, a total of 512 possible color combinations (8 x 8 x 8) are selectable for each entry.

Mode 00 (320x200x4) can index all sixteen palette colors; while mode 01 (640x200x2) can index just the first four (Reg0 - Reg3) palette colors. The monochrome mode (10 - 640x400x1) bypasses the color palette and is instead provided with an inverter for inverse video controlled by bit 0 of palette color 0 (ST Reg 0). Color palette 0 is also used to assign a border color while in multi-plane mode.

Additional resolution modes are available by programming the shifter through the TT Shift Mode register. In these modes, there are a maximum of 256 TT Color Palette Registers each containing 12-bits of color: 4-bits each for red, green, and blue. Therefore, a total of 4096 possible color combinations (16 x 16 x 16) are selectable. Through the ST Palette Bank (lowest 4 bits of the TT Shift Mode Register) one of 16 banks may be selected from the TT Color Palette for use in ST modes. This allows modes 000, 001, 010, and 100 to seemingly select from up to 256 registers by simply altering the palette bank. Only mode 111 (320x480x8) can index [illegible] registers.

4.2. Video RAM/Controller/Display Interface

Video display memory is configured as logical planes (1, 2, 4, or 8) of interwoven 16-bit words of contiguous memory to form one 32,000 byte (for ST modes) or 153,600 byte (for TT modes) physical plane starting at any 8 byte boundary (in dual-purpose RAM only). The starting address of display memory is loaded into the Video Base High, Video Base Mid, or Video Base Low Registers (the most significant byte of the thirty two bit addresses is always zero, i.e. within the ST image). This register is loaded into the Video Address Counter (High/Mid/Low) at the beginning of each frame. The address counter is incremented as the BitMap planes are read.

BitMap planes are transferred to the video chip (TT shifter) buffer 64-bits at a time. The shifter then loads the video shift register where one bit from each plane is shifted out and collectively used as the index (plane 0 appears first in RAM and provides the least significant bit of each pixel) to a specific ST or TT Palette Register (depending on the Shift Mode).

4.3. Monitor Connector

The video output is provided on a 3 row 15 pin connector similar to the one used on the Atari PC4 VGA.

Pin	Function
1	Red
2	Green
3	Blue
4	High Resolution Monochrome Out +
5	Ground
6	Red Return
7	Green Return
8	Blue Return
9	Monochrome Detect (input)
10	Ground
11	Open
12	Open
13	Hsync
14	Vsync
15	High Resolution Monochrome Out -

5. *Music Subsystem*

The TT architecture extends the music subsystem presently available on the ST/MEGA computers. The TT mixes the output of the existing ST PSG sound system with a new DMA-driven dual-channel D-to-A subsystem. The TT includes an internal speaker driven by these two sources for simple beeps, and can be connected to an external stereo amplifier for high-fidelity sound.

The TT is also equipped with a Musical Instrument Digital Interface (MIDI) which provides high speed serial communication of musical data to and from more sophisticated synthesizer devices.

5.1. *Programmable Sound Generator*

The ST sound system using the General Instruments AY-3-8910 / Yamaha YM-2149 Programmable Sound Generator is present in the TT. The YM-2149 Programmable Sound Generator produces music synthesis, sound effects, and audio feedback. With an applied clock input of 2 MHz, the PSG is capable of providing a frequency response range between 30 Hz (audible) and 124 KHz (post-audible). The generator places minimal amount of processing burden on the main system (which acts as the sequencer) and has the ability to perform using three independent voice channels. The three sound channel outputs are mixed together sent to the volume and tone control chip.

(Reference *Engineering Hardware Specification of the Atari ST Computer System*, page 10.)

5.2. *DMA Sound*

The TT also includes a new DMA-driven sound subsystem that allows the playback or synthesis of complex waveforms at a variety of sampling rates.

5.2.1. *Overview*

Sound in the form of digitized samples is stored in system memory. These samples are fetched from dual-purpose memory during horizontal blanking (transparent to the processor) and provided to a digital-to-analog converter (DAC) at a constant sample frequency specified by the user. The output of DAC is then low pass filtered to a frequency equal to forty percent of the sample frequency by a four pole switched capacitor low pass filter. The signal is further filtered by a two pole fixed frequency (15 kHz) low pass filter and provided to a National LMC1992 Volume / Tone Controller. Finally, the output of this device is available at a pair of RCA jacks and an internal speaker.

Two channels are provided. They are intended to be used as the left and right channels of a stereo system when using the raw audio outputs from the machine. Of course, they are mixed together when fed to the internal speaker. A mono mode is provided which will feed the same data to both channels simultaneously. The only restriction placed on mono mode is that there must be an even number of samples (see data format section for details).

5.2.2. *Data Format*

Each sample is stored as an eight bit quantity, the most significant bit is the sign and the other seven bits are magnitude. In the stereo scheme there is one word per sample, the upper byte contains the left channel sample and the lower byte contains the right channel sample. In the mono scheme bytes are accessed sequentially. However, they are still fetched a word at a time. Therefore, there must be an even number of samples.

A group of samples is called a frame. A frame may be played once or can automatically be repeated forever. Frames occupy a contiguous block of memory and are specified by their starting and ending addresses. The ending address is the address of the last sample + 2. An external clock is provided to timer A of the ST MFP at the end of each frame. This can be used as an interrupt. This pulse is also exclusive OR'ed with the monochrome monitor detect bit, whose transistion can generate an interrupt on bit 7 of the MFP-ST General Purpose I/O Port. Frames may be linked together by defining a new frame while the current frame is being played. The new frame will begin at the end of the current frame.

As an example, suppose you have three frames (A, B, and C) and we want to play frame A once, then play frame B 5 times, and finally play frame C twice. To accomplish this you can do the following:

1. Setup frame A.
2. Write 3 to the sound DMA control register to start playing with repeat.
3. Setup timer A to use an external clock, initialize its count to 5, and have it interrupt when count = 0.
4. Setup frame B.
5. Go do something else until interrupted.
6. Setup frame C.
7. Setup timer A count to 2.
8. Go do something else until interrupted.
9. Write 1 to the sound DMA control register to cause playing to stop at the end of the frame.

In this example no mention is made of setting the sample rate, volume or tone controls. It's assumed that all of these have been set up ahead of time. It should be obvious how this example can be extended to allow volume or tone to be modified at specific points during playback.

Note If we had loaded the sound DMA control register with a 1 in step 2, frame A would have been played once and sound would have been disabled. A zero can be written to the sound DMA control register at any time to stop playback immediately.

5.2.3. *MICROWIRE Interface*

The MICROWIRE interface provided to talk to the National LMC1992 Computer Controlled Volume / Tone Control is a general purpose MICROWIRE interface to allow the future addition of other MICROWIRE devices. For this reason, the following description of its use will make no assumptions about the device being addressed.

The MICROWIRE bus is a three wire serial connection and protocol designed to allow multiple devices to be individually addressed by the controller. The length of the serial data stream depends on the destination device. In general, the stream consists of N bits of address, followed by zero or more don't care bits, followed by M bits of data. The hardware interface which has been provided consists of two 16 bit read/write registers. One data register which contains the actual bit stream to be shifted out and one mask register which indicates which bits are valid.

Let's consider a mythical device which requires two address bits and one data bit. For this device the total bit stream is three bits (minimum). Any contiguous three bits of the register pair may be used. However, since the most significant bit is shifted first, the command will be received by the device soonest if the three most significant bits are used. Let's assume: 01 is the device's address, D is the data to be written, and X's are don't cares. Then all of the following register combinations will provide the same information to the device.

```
1110 0000 0000 0000  Mask
01DX XXXX XXXX XXXX  Data

0000 0000 0000 0111  Mask
XXXX XXXX XXXX X01D  Data

0000 0001 1100 0000  Mask
XXXX XXX0 1DXX XXXX  Data

0000 1111 1111 0000  Mask
XXXX 01XX XXXD 0000  Data

1111 1111 1111 1111  Mask
01XX XXXX XXXX XXXD  Data
```

The mask register needs to be written before the data register. Sending commences when the data register is written and takes approximately 16uS. Subsequent writes to the data and mask registers are blocked until sending is complete. Reading the registers while sending is in progress will return a snapshot of the shift register shifting the data and mask out. This means that you know it is safe to send the next command when these registers (or either one) return to their original state. Note that the mask register does not need to be rewritten if it is already correct. That is, when sending a series of commands the mask register only needs to be written once.

5.2.4. *Volume and Tone Control*

The LMC1992 is used to provide volume, tone, and mixing control. This part is talked to using the MICROWIRE interface. The device has a two bit address field, address = %10, and a nine bit data field. There is no way of reading the current settings.

The input selector is used to enable and disable mixing the output of the GI PSG with the DMA sound. After reset, the input is grounded, and should be switched to either states 1 or 2 during initialization to avoid level mismatches during later switching.

Data Field

```
011 DDD DDD  Set Master Volume
    ||| |||
    000 000  -80 dB
    010 100  -40 dB
    101 XXX    0 dB

101 XDD DDD  Set Left Channel Volume
     || |||
     00 000  -40 dB
     01 010  -20 dB
     10 1XX    0 dB

100 XDD DDD  Set Right Channel Volume
     || |||
     00 000  -40 dB
     01 010  -20 dB
     10 1XX    0 dB

010 XXD DDD  Set Treble
      | |||
      0 000  -12 dB
      0 110    0 dB (Flat)
      1 100  +12 dB

001 XXD DDD  Set Bass
      | |||
      0 000  -12 dB
      0 110    0 dB (Flat)
      1 100  +12 dB

000 000 0ss  GI PSG Sound Enable
          ||
          00  disabled, unbiased
              (reset state)
          01  enabled
          10  disabled, biased
```

Note: The volume controls attenuate in 2 dB steps. The tone controls attenuate in 2 dB steps at 100 Hz and 15 kHz.

5.3. *Musical Instrument Digital Interface (MIDI)*

The MIDI allows the integration of the TT series with music synthesizers, sequencers, drum boxes, and other devices possessing MIDI interfaces. High speed (31.25 Kbaud) serial communication of keyboard and program information is provided by two ports, MIDI OUT and MIDI IN (the MIDI OUT also includes MIDI THRU data).

The MIDI communicates through the MC6850 Asynchronous Communications Interface Adapter (ACIA) to the system bus. The data transfer rate is a constant 31.25 Kbaud of 8-bit asynchronous data.

(Reference *Engineering Hardware Specification of the Atari ST Computer System*, pages 11 and 17 for more information on the MIDI and ACIA.)

6. *VMEbus*

The TT and TT/X provide for I/O expansion and by implementing the industry standard VMEbus, revision C.1. The TT/X can also accommodate alternate bus masters such as multiple processors. The TT has one single-high VMEboard backplane. The TT/X's configuration is a 5 slot, double-high VMEboard backplane.

6.1. *System Controller*

The main system board serves as the VMEbus system controller (a slot 1 "card") and implements the following functions:

- single-level (level-three) VMEbus arbiter
- IACK* daisy-chain driver
- global SYSCLK (16 MHz, independent of processor speed)
- global VMEbus time-out that drives BERR*

The level-three arbiter is designed to meet the VMEbus specification requirements.

The IACK* daisy-chain driver is designed to meet the VMEbus specification requirements.

The SYSRESET* line is driven low when (1) power-up occurs, (2) the reset pushbutton is depressed, or (3) the 68030 asserts its RESET* signal.

6.2. *Address Partitioning*

The starting address of the VME address space as seen by the 68030 in TT/X can be configured to be contiguous with the top of the single-purpose fast system RAM (by straps). Part of the 32-bit wide physical address space is partitioned off to provide VME A24 and A16 address spaces.

The TT's A24/D16 VMEbus interface is fixed at the same location as the TT/X A24/D16 space: 0xFE000000-0xFEFFFFFF.

6.3. *Read-Modify-Write Cycles*

The bus can not be arbitrated away from the 68030 if it is in the midst of a read-modify-write cycle.

6.4. *VME Interrupter*

The system can write to an I/O address to generate a level 3 interrupt on the VMEbus. It can monitor a status register that indicates when that interrupt has been acknowledged and serviced. An I/O address contains a read/write status/control port, only the least significant bit of the least significant byte is defined. When set to 1, it generates a VMEbus level 3 interrupt. When cleared, the interrupt request is taken away.

Note that the level 3 interrupt must be masked off (either by setting the processor's IPL or by masking the interrupt in the system controller) or the 68030 will be immediately interrupted.

The system board responds to a VMEbus interrupt acknowledge cycle with the status ID of 0xFF.

7. I/O Expansion

7.1. Ethernet (tm)

To support connectivity in the established office and workstation environments it is imperative that there be a relatively low cost Ethernet board. Unfortunately, all VME based Ethernet controllers are priced over $2,000. Spurred by this unacceptable high cost, various low-cost possibilities are being considered: 1) an Atari-developed ST compatible ACSI/DMA Ethernet controller, 2) an Atari-developed VME/Ethernet controller, and 3) the Adaptec SCSI/LAN Interface controller -- Nodem.

For the TT/UNIX system, the short term strategy uses the Adaptec Nodem for the Ethernet connection. The VME bus Ethernet controller is targeted as a future product.

The TT/TOS system may use the ACSI/DMA Ethernet controller being developed for the ST and MEGA computer series.

7.2. Multi-Port Serial Card

To provide for people that want multiple users on a single system, a low cost multiple port serial card that is capable of supporting at least 8 RS-232C serial lines, each operating at up to 38.4 Kbps, should be provided.

7.3. Workstation Video Subsystem

The TT can also be configured as a workstation by plugging in a high resolution video engine VMEbus card and taking advantage of the integral keyboard and mouse interfaces. The workstation video display subsystem is based on the Perihelion video architecture introduced in the Atari Transputer Workstation (ATW). It includes a custom hardware ByteBlt capability that allows it to become a VMEbus bus master.

A variety of video resolutions are provided including:

- 1280 x 960 x 4 bits/pixel (Mode 0)
- 1024 x 768 x 8 bits/pixel through a color lookup table (Mode 1)
- 640 x 480 x 8 bits/pixel through a color lookup table (2 screens) (Mode 2)
- 512 x 480 x 32 bits/pixel (24 bits true color plus overlay and tag bits) (Mode 3)

The display is implemented with 32 bit wide video RAMs, providing direct VMEbus access to the display RAM. Additionally the ByteBlt hardware is capable of becoming a VMEbus master and doing transfers into and out of system RAM. ByteBlt transfers within the video RAM can take place without gaining control of the VMEbus. The video subsystem is also capable of generating an interrupt on the VME bus.

The video display RAM appears multiple times in the VME address space to facilitate processor accesses on pixel boundaries. It appears

in the obvious double word width format, made up of four independent bytes. It is also has two additional images that put the nibbles of each byte in the least significant nibble of each byte of a double word wide access. This makes it easy for the processor to access individual pixels without shifting or masking when in the four bits per pixel mode. One image would contain all of the least significant nibbles and the other image all of the most significant nibbles.

The video subsystem has 42 double word wide read/write control and status registers.

8. *SYSTEM*

8.1. *Boot Sequence*

The TT/X ROM will contain power-on diagnostics to verify that the processor, memory, and I/O subsystems are functional. The boot sequence begins after these diagnostic tests are successfully completed.

The boot process has three general stages:

1) The ROM boot procedure searches peripherals for boot code. This determines the order that various peripherals will be searched for code.

2) The device boot is loaded by the ROM boot. Where the device is known, *device* is replaced by the device from which the boot was loaded, e.g., the boot loaded from the hard disk is referred to an the "hard disk boot", or simply, "disk boot". The device boot consists of 512 bytes of boot code from the "boot sector" of the device and is loaded at a known point in dual-purpose RAM (see Section 2.3). Some devices, such as hard disks, load a second sector of boot code that calls code in the first sector.

3) The UNIX boot is loaded by the device boot. It is typically a moderately sized program (32 to 64 Kbytes) that actually loads the UNIX operating system. It need not be position independent if its location has been agreed upon with the device boot.

Note that in TOS systems, the device boot typically loads the operating system itself, rather than another boot program, so step 3 is omitted. However, the device boot may first load in another sector of boot code from its boot partition.

Them ROM boot procedure's main purpose is to detect boot devices and load and run the device boot code on these devices. It checks the following devices in order:

1) Cartridge

2) Floppy drive 0

3) ACSI hard disk drives

4) SCSI hard disk drives

5) Ethernet

6) ROM

For detailed information on the TT boot procedures see the *Atari TT Boot Specification.*

8.2. *Operating System*

The TT/X is intended for use with the UniPlus+ V.3.1 operating system supplied from Unisoft. The implementation is based on release 3.1 of the AT&T System V operating system. It conforms to the AT&T System V Interface Definition, [illegible] standards, and X/OPEN internationalization standard. UniPlus features also include the fast file system and socket

architecture of 4.3BSD.

8.3. *Device Drivers*

UniPlus+ is supplied with configuration tools which allow device drivers to be fully configurable with the UNIX kernel.

8.4. *Networking Support*

To permit its use in a wide variety of environments, the TT series of computers has software support for the Ethernet network, the Internet networking protocols (TCP/IP), and Sun Microsystem's Network File System (NFS).

8.5. *Windowing User Interface*

The operating system will include a windowing user interface built on the X-Windows (Version 11.3) package.

8.6. *Binary Compatibility Standard*

Systems equipped with the 3.5" 1.44 Mbyte tape drives are capable of reading and writing Motorola/Unisoft Binary Compatibility Standard (BCS) floppy media. This media is defined as IBM PS/2 3.5" Floppy Disk Format.

Systems equipped with the Archive VIPER 5.25" SCSI streaming cartridge tape drive shall be able to read the BCS compatible QIC-24 formatted tapes. These drives are not capable of writing QIC-24 formatted media.

9. *Mechanical Considerations*

9.1. TT System Packaging

The TT system is packaged as a desktop product using a plastic cabinet originally designed for the Atari ST+. This provides sufficient internal space one 3.5" floppy, a hard disk, ST RAM memory expansion,one single height VME card, and possibly 4Mb of FAST RAM expansion.

9.2. TT/X System Packaging

The TT/X system is contained in a floor-standing tower. This desk-side packaging also includes 4 Mbyte of "system" RAM as standard, and a 5-slot double-Eurocard VME cardcage.

9.3. Power Supply

The power supply used in the TT will be capable of delivering 53W, including at least 7 amps of +5VDC, at least 1 amp of +12VDC (with allowance for up to 2.2A at power-up for 1 second), at least 300 milliamps of -12VDC, and at least 400 milliamps of -5VDC. The standard power supply used to power the TT/X will be capable of delivering 200 W, including at least 20 amps of +5VDC, at least 8 amps of +12VDC, at least 300 milliamps of -12VDC, and at least 250 milliamps of -5VDC. The supply will also generate a "power good" signal that is asserted after the supply voltages are stable, and is removed before the supply voltages are removed.

The *size* field has the following designations:

DW Double word

W Word wide

OB Odd byte (A byte wide port that appears in the least significant byte of the defined words. The most significant byte of the words is undefined. If desired, these ports may be accessed as bytes by adding 1 to the specified word addresses.)

EB Even byte (A byte wide port that appears in the most significant byte of the defined words. The least significant byte of the words is undefined.)

10. Memory, I/O, & Interrupt Map

MEMORY MAP as seen by the 68030

address	size	cache-able	use
00000000-00EFFFFF	DW	yes	ST (dual-purpose) RAM, ROM
00F00000-00F7FFFF	W	no	<reserved TT I/O>
00F80000-00FFFFFF	W	no	ST & TT IO
01000000-01xxxxxx	DW	yes	TT fast RAM (opt.) } 16 Mo FRAM
01xxxxxx-01FFFFFF	DW	yes	~~strappable start of VME~~ } 16 Mo FRAM
02000000-7FFFFFFF	DW	yes	A32:D32 VME RAM (Expansion)
80000000-BFFFFFFF	DW	no	A32:D32 Memory/Peripherals
C0000000-FCFFFFFF	W	no	A32:D16 Memory/Peripherals
FD000000-FDFFFFFF	DW	no	VMEbus A24:D32
FE000000-FEFEFFFF	W	no	VMEbus A24:D16
FEFF0000-FEFFFFFF	W	no	VMEbus A16:D16
FF000000-FFFFFFFF	--	--	ST compatible image (a write to FFD000xx sets the single-purp

st RAM refresh rate; and

simultaneously generates a bus error)

ST Compatible Image (Base Address 00000000 OR FF000000)

address	size	cache-able	use
000000-000007	DW	yes	ROM (image of first 8 bytes of main ROM, supervisor mode, read only)
000008-9FFFFF	DW	yes	"dual-purpose" RAM (memory in the range 000008-0007FF is only accessible in supervisor mode)
A00000-DFFFFF	-	yes	~~<reserved>~~ VME 4Mo (MSTE / TT)
E00000-EFFFFF	DW	yes	Main ROM
F00000-F9FFFF	-	no	<reserved>
FA0000-FBFFFF	W	no	Cartridge ROM

E00000/E40CFE

FC0000-FF7FFF	-	no	<reserved>
FF8000-FFFFFF	W	no	ST & TT I/O Space

ST/TT I/O MAP (Offset within ST image FF8000)
(Base Address 00FF8000 OR FFFF8000).

offset	size	use
8000-8001	OB	Memory Controller
8002-81FF	-	<reserved>
8200-8263	OB	TT Video Subsystem
8264-83FF	-	<reserved>
8400-85FF	W	TT Palette
8600-86FF	W	ST DMA and FDC
8700-8715	OB	SCSI DMA Control
8716-877F	-	<reserved>
8780-878F	OB	SCSI Controller
8790-87FF	-	<reserved>
8800-8803	EB	ST Sound Chip
8804-88FF	-	<reserved>
8900-893F	OB	DMA Sound Control
8940-895F	-	<reserved>
8960-8963	OB	Real Time Clock and NVRAM
8964-8BFF	-	<reserved>
8C00-8C15	OB	SCC DMA Control
8C16-8C7F	-	<reserved>
8C80-8C87	OB	SCC
8C88-8DFF	-	<reserved>
8E00-8E1F	OB	System Control Unit (SCU)
8E20-91FF	-	<reserved>
9200-9201	EB	Configuration Switches
9202-9FFF	-	<reserved>
A000-A3FF	W	TT main board peripheral expansion
A400-F9FF	-	<reserved>
FA00-FA3F	OB	MFP-ST
FA40-FA7F	-	<reserved>
FA80-FABF	OB	MFP-2
FAC0-FBFF	-	<reserved>
FC00-FC03	EB	IKBD Interface
FC04-FC07	EB	MIDI ACIA
FC08-FFFF	-	<reserved>

LOCAL I/O DEVICES

MEMORY CONTROLLER

8000 W Memory Configuration (Note 1)
(
B7-B4 <reserved>
B3-B2 <reserved, R/W latches>
B1 0 256 Kbyte parts
1 1 Mbyte parts
B0 <reserved, R/W latch>
)

ST/TT VIDEO SUBSYSTEM

8200	RW	---- ----	xxxx xxxx	Video Base High
8202	RW	---- ----	xxxx xxxx	Video Base Mid
8204	RO	---- ----	xxxx xxxx	Video Address Counter High
8206	RO	---- ----	xxxx xxxx	Video Address Counter Mid
8208	RO	---- ----	xxxx x000	Video Address Counter Low
820A	RW	---- --0x		ST Sync Mode (set to 1)
820B	WO		0000 0000	<reserved>
820C	RW	---- ----	xxxx x000	Video Base Low
8240	RW	---- -rrr	-ggg -bbb	ST Color Palette Reg0
8242	RW	---- -rrr	-ggg -bbb	ST Color Palette Reg1
825E	RW	---- -rrr	-ggg -bbb	ST Color Palette Reg15
8260	RW	---- --ss	---- ----	ST Shift Mode (ss 00 320x200, 4 plane 01 640x200, 2 plane 10 640x400, 1 plane 11 <reserved>)
8262	RW	s--h -mmm	---- bbbb	TT Shift Mode (s sample and hold mode) (h hyper mono mode) (mmm 000 320x200x4 001 640x200x2 010 640x400x1 100 640x480x4 110 1280x960x1 111 320x480x8) (bbbb ST palette bank)

TT VIDEO SUBSYSTEM

8400	RW	---- rrr	[illegible]	TT Palette Reg0
8402	RW	---- rrr	[illegible]	TT Palette Reg1

...

85FE	RW	---- rrrR	gggG bbbB		TT Palette Reg255

ST DMA

8600				<reserved>
8602				<reserved>
8604	RW	---- ----	xxxx xxxx	Disk Data Path (WDC)
8606	RO	---- ----	---- -xxx	DMA Status
8606	WO	---- ---x	xxxx xxxx	DMA Mode (WDL)
8608	RW	---- ----	xxxx xxxx	DMA Pointer High
860A	RW	---- ----	xxxx xxxx	DMA Pointer Mid
860C	RW	---- ----	xxxx xxx0	DMA Pointer Low
860E	RW	---- ----	0000 00dc	Floppy Density Select (d - FDDS (output) pin 0 = low (reset) 1 = high) (c - FCCLK pin 0 = 8MHz (reset) 1 = 16MHz)

DMA SCSI1

8700	RW	---- ----	xxxx xxxx	DMA Pointer Upper
8702	RW	---- ----	xxxx xxxx	DMA Pointer Upper-Middle
8704	RW	---- ----	xxxx xxxx	DMA Pointer Lower-Middle
8706	RW	---- ----	xxxx xxxx	DMA Pointer Lower
8708	RW	---- ----	xxxx xxxx	Byte Count Upper
870A	RW	---- ----	xxxx xxxx	Byte Count Upper-Middle
870C	RW	---- ----	xxxx xxxx	Byte Count Lower-Middle
870E	RW	---- ----	xxxx xxxx	Byte Count Lower
8710	RO	xxxx xxxx	xxxx xxxx	Data Residue Register High
8712	RO	xxxx xxxx	xxxx xxxx	Data Residue Register Low
8714	RW	---- ----	bz00 00ed	Control Register (b - bus error during DMA (read only, cleared by read) z - byte count zero (read only, cleared by read) e - DMA enable 0=off, 1=on d - DMA direction: 0=in from port 1=out to port)

SCSI Controller

8780	OB	Data Register
8782	OB	Initiator Command Register

8784	0B	Mode Register
8786	0B	Target Command Register
8788	0B	ID Select/SCSI Control Register
878A	0B	DMA Start/DMA Status Register
878C	0B	DMA Target Receive/Input Data
878E	0B	DMA Initiator Receive/Reset

PROGRAMMABLE SOUND GENERATOR
(also provides bi-directional parallel printer port and miscellaneous output latch)

8800	RO	xxxx xxxx	---- ----	PSG Read Data
8800	WO	0000 xxxx	---- ----	PSG Register Select
8802	WO	xxxx xxxx	---- ----	PSG Write Data

Port A Bit Assignments

7	*LAN Select (0 routes SCC Port A to LAN connector)
6	~~<reserved>~~ speaker on/off
5	Printer Port Strobe
4	*DTR (MFP-ST serial port)
3	*RTS (MFP-ST serial port)
2	*Floppy 1 Select
1	*Floppy 0 Select
0	*Floppy Side 0 Select

Port B Bit Assignments

7-0	Printer Port bits 7-0

DMA SOUND SUBSYSTEM

8900	RW	---- ----	0000 00re	Sound DMA Control (r - Repeat 0 = Single Frame 1 = Repeat e - Enable 0 = Off (reset state) 1 = On)
8902	RW	---- ----	xxxx xxxx	Frame Base Address (high)
8904	RW	---- ----	xxxx xxxx	Frame Base Address (med)
8906	RW	---- ----	xxxx xxxx	Frame Base Address (low)
8908	RW	---- ----	xxxx xxxx	Frame Address Counter (high)
890A	RW	---- ----	xxxx xxxx	Frame Address Counter (med)
890C	RW	---- ----	xxxx xxxx	Frame Address Counter (low)
890E	RW	---- ----	xxxx xxxx	Frame End Address (high)
8910	RW	---- ----	xxxx xxxx	Frame End Address (med)

```
8912    RW    ---- ----  xxxx xxxx   Frame End Address (low)

8920    RW    0000 0000  a000 00bb   Sound Mode Control
                                      (
                                      a - Mode
                                          0 = Stereo (reset state)
                                          1 = Mono
                                      bb - Sample Rate
                                          00 =  6258 Hz
                                          01 = 12517 Hz
                                          10 = 25033 Hz
                                          11 = 50066 Hz
                                      )
8922    RW    xxxx xxxx  xxxx xxxx   MICROWIRE Data register
8924    RW    xxxx xxxx  xxxx xxxx   MICROWIRE Mask register

REAL TIME CLOCK (MC146818A)

8960    0B    Real Time Clock Address Register
8962    0B    Real Time CLock Data Register

DMA SCC

8C00    RW    ---- ----  xxxx xxxx   DMA Pointer Upper
8C02    RW    ---- ----  xxxx xxxx   DMA Pointer Upper-Middle
8C04    RW    ---- ----  xxxx xxxx   DMA Pointer Lower-Middle
8C06    RW    ---- ----  xxxx xxxx   DMA Pointer Lower

8C08    RW    ---- ----  xxxx xxxx   Byte Count Upper
8C0A    RW    ---- ----  xxxx xxxx   Byte Count Upper-Middle
8C0C    RW    ---- ----  xxxx xxxx   Byte Count Lower-Middle
8C0E    RW    ---- ----  xxxx xxxx   Byte Count Lower

8C10    RO    xxxx xxxx  xxxx xxxx   Data Residue Register High
8C12    RO    xxxx xxxx  xxxx xxxx   Data Residue Register Low

8C14    RW    ---- ----  bz00 00ed   Control Register
                                      (
                                      b - bus error during DMA
                                          (read only, cleared by read)
                                      z - byte count zero
                                          (read only, cleared by read)
                                      e - DMA enable 0=off, 1=on
                                      d - DMA direction:
                                           0=in from port
                                           1=out to port
                                      )

8530 SCC

8C80    0B    SCC1 A control
8C82    0B    SCC1 A data
```

8C84	0B	SCC1 B control
8C86	0B	SCC1 B data

SCU

8E00	0B	System Interrupt Mask (B7 - B1; B0 unused)
8E02	0B	System Interrupt State (read only; before mask register)
8E04	0B	System Interrupter (B0 = generate interrupt 1)
8E06	0B	VME Interrupter (B0 = generate interrupt VME IRQ3)
8E08	0B	SCU General Purpose Register 1 (reset only at power-up)
8E0A	0B	SCU General Purpose Register 2 (reset only at power-up)
8E0C	0B	VME Interrupt Mask (B7 - B1; B0 unused)
8E0E	0B	VME Interrupt State (read only; before mask register)

Configuration Switch Register

9200	EB	ID Switches (Read Only)

MFP-ST (ST compatible)

FA00	0B	GPIP
FA02	0B	AER
FA04	0B	DDR
FA06	0B	IERA
FA08	0B	IERB
FA0A	0B	IPRA
FA0C	0B	IPRB
FA0E	0B	ISRA
FA10	0B	ISRB
FA12	0B	IMRA
FA14	0B	IMRB
FA16	0B	VR
FA18	0B	TACR
FA1A	0B	TBCR
FA1C	0B	TCDCR
FA1E	0B	TADR
FA20	0B	TBDR
FA22	0B	TCDR
FA24	0B	TDDR
FA26	0B	SCR
FA28	0B	UCR
FA2A	0B	RSR
FA2C	0B	TSR
FA2E	0B	UDR

MFP2

FA80	0B	GPIP
FA82	0B	AER
FA84	0B	DDR
FA86	0B	IERA
FA88	0B	IERB

FA8A	OB	IPRA
FA8C	OB	IPRB
FA8E	OB	ISRA
FA90	OB	ISRB
FA92	OB	IMRA
FA94	OB	IMRB
FA96	OB	VR
FA98	OB	TACR
FA9A	OB	TBCR
FA9C	OB	TCDCR
FA9E	OB	TADR
FAA0	OB	TBDR
FAA2	OB	TCDR
FAA4	OB	TDDR
FAA6	OB	SCR
FAA8	OB	UCR
FAAA	OB	RSR
FAAC	OB	TSR
FAAE	OB	UDR

ikbd ACIA

FC00	EB	Keyboard ACIA Control
FC02	EB	Keyboard ACIA Data

MIDI ACIA

FC04	EB	MIDI ACIA Control
FC06	EB	MIDI ACIA Data

Note 1: A pin on the memory controller can be strapped to force the memory controller to ignore the configuration register and automatically use 256K part mode. This is used for the second of two memory controllers. The configuration register still applies to the primary memory controller. This allows 2Mb or 8Mb connected to the primary memory controller, and 2Mb to the secondary controller for possible dual-purpose RAM configurations of 2Mb, 4Mb, 8Mb, or 10Mb.

Note 2: Two TT glue chip pins, IOCS1 and IOCS2, output the decode of offsets within the I/O area of 0xA000-0x00A1FF and 0xA200-0xA3FF, respectively. These pins minimize decoding when adding peripherals to the TT main board sometime in the future.

VME ADDRESS SPACE

ADDRESS SPACE SEEN BY A32 VME BUS MASTER
(logically equal to that seen by 68030, but without the address modifiers and size constraints)

address	size	use
00000000-00EFFFFF	DW	ST RAM, ROM
00F00000-00FFFFFF	W	TT IO
01000000-01xxxxxx	DW	TT fast RAM (opt.)
01xxxxxx-01FFFFFF	DW	strappable start of VME
02000000-FEFFFFFF	DW,W,B	VME Expansion
FF000000-FFFFFFFF	DW,W,B	ST Image

ADDRESS SPACE SEEN BY A24 VME BUS MASTER
(sees only the ST Image)

address	size	use
000000-EFFFFF	DW	ST ROM/RAM
F00000-FFFFFF	W	TT IO

VME CONTROLLER STARTING ADDRESSES
(Three pins strapped on the VME controller giving 8 starting addresses for VME space)

MS2	MS1	MS0	address
0	0	0	01000000
0	0	1	01100000
0	1	0	01200000
0	1	1	01300000
1	0	0	01400000
1	0	1	01800000
1	1	0	01C00000
1	1	1	02000000

FE000000-FEFEFFFF VME A24 / D16

INTERRUPT ASSIGNMENTS

int	system	vector	VME	vector
7	VMEbus SYSFAIL	AutoVector	IRQ7	programmable
6	none	-	MFPs & IRQ6	programmable
5	none	-	SCC & IRQ5	programmable
4	VSYNC	AutoVector	IRQ4	programmable
3	(Note 3)	-	VME Interrupter & IRQ3	AutoVector & programmat
2	HSYNC	AutoVector	IRQ2	programmable
1	System Interrupter	AutoVector	IRQ1	programmable

Note 1: Within each level, the system interrupt has higher priority than the VME interrupt. And, within the shared Level5 and Level6 interrupts, the part on the motherboard has higher priority than the VME interrupt.

Note 2: The VME interrupts use their interrupt status byte as their interrupt vector.

Note 3: The level 3 system interrupt mask must be enabled for the level 3 VME interrupt to actually be generated.

MFP Interrupt Assignments

MFP-ST (ST Compatible)	
int	function
GPIP7	Monochrome Monitor Detect / DMA Sound IRQ
GPIP6	Ring Indicator
TimerA	
RxRDY	
RxERR	
TxEMPTY	
TxERR	
TimerB	
GPIP5	ACSI / FDC Interrupt
GPIP4	MIDI / Keyboard Interface
TimerC	
TimerD	
GPIP3	<reserved>
GPIP2	CTS
GPIP1	DCD
GPIP0	Centronics BUSY

MFP 2	
int	function
GPIP7	SCSI Controller IRQ (active high)
GPIP6	RTC IRQ (active low, cleared by reading RTC register
TimerA	
RxRDY	
RxERR	
TxEMPTY	
TxERR	
TimerB	
GPIP5	SCSI DMAC Interrupt (active low)
GPIP4	Diskette ChangeLine
TimerC	
TimerD	
GPIP3	Ring Indicator (SCC B)
GPIP2	SCC DMAC Interrupt (active low)
GPIP1	general purpose I/O pin
GPIP0	general purpose I/O pin

DMA/BUS MASTERSHIP PRIORITIES

priority	function
highest	ACSI/Floppy Controller
	SCC DMA Controller
	SCSI DMA Controller
	VMEbus Masters
lowest	68030

11. *Prototype Differences*

The items in this section are unique to the prototype TT and the developers' machines known as the TT/D.

11.1. *ID Switches*

The prototype TT systems do NOT have an 8 bit ID/Configuration switch port. On TT/D systems built with MCU-B, the port will be in the most significant 8 bits of I/O offset 0x8000. On TT/P systems the port is at its standard I/O offset of 0x9200.

11.2. *Network*

The prototypes include provision for *PromiseLAN* which will not be present in the final units. Also, there are two sets of jumpers that are used to select the format of SCC Port A output instead of being under software control.

11.3. *Cartridge Port*

The cartridge port is not easily accessible on the prototype machines (the cabinet must be opened).

11.4. *VMEbus*

If the VMEbus backplane is not connected, the external interrupt lines should be pulled up on the motherboard. (In practice, this only involves XIRQ6 and XIRQ5 because of their shared nature with the motherboard "system" interrupts.)

11.5. *SCC and SCSI DMA Controllers*

In the prototypes, the SCC and SCSI DMA controllers do not automatically disable DMA when their respective byte counters reach zero. They do generate an interrupt when the byte counter reaches zero.

11.6. *RAM Expansion*

The TT/D is limited to only 2 of the 4 Mb single purpose RAM expansion cards, for a total of 8 Mb of single purpose RAM.

11.7. *SCU*

The pushbutton reset does not reset the registers in the SCU on either the prototype TT or the TT/D.